Project 1

Game: 10,000

Course: CIS-5

Section Number: 41366

Name: Scott Renfrow

Due Date: February 7, 2020

Introduction: For the dice game 10000, the object of the game is to be the first to get 10,000 points. This game is played with 6 dice that are all rolled at the same time. A player gains points by rolling certain combinations of die. Each player rolls the dice once, adds up their points, then passes the dice to the next player.

How Points are Added:

1. Every time a 1 is rolled, 100 points are added to your score.
2. Every time a 5 is rolled, 50 points are added to your score.
3. When a 3 of a kind is rolled the number on the die is multiplied by 100 and added to your score. For example, if a player rolls three 4's, that player adds 400 to their total. One exception to these points is if

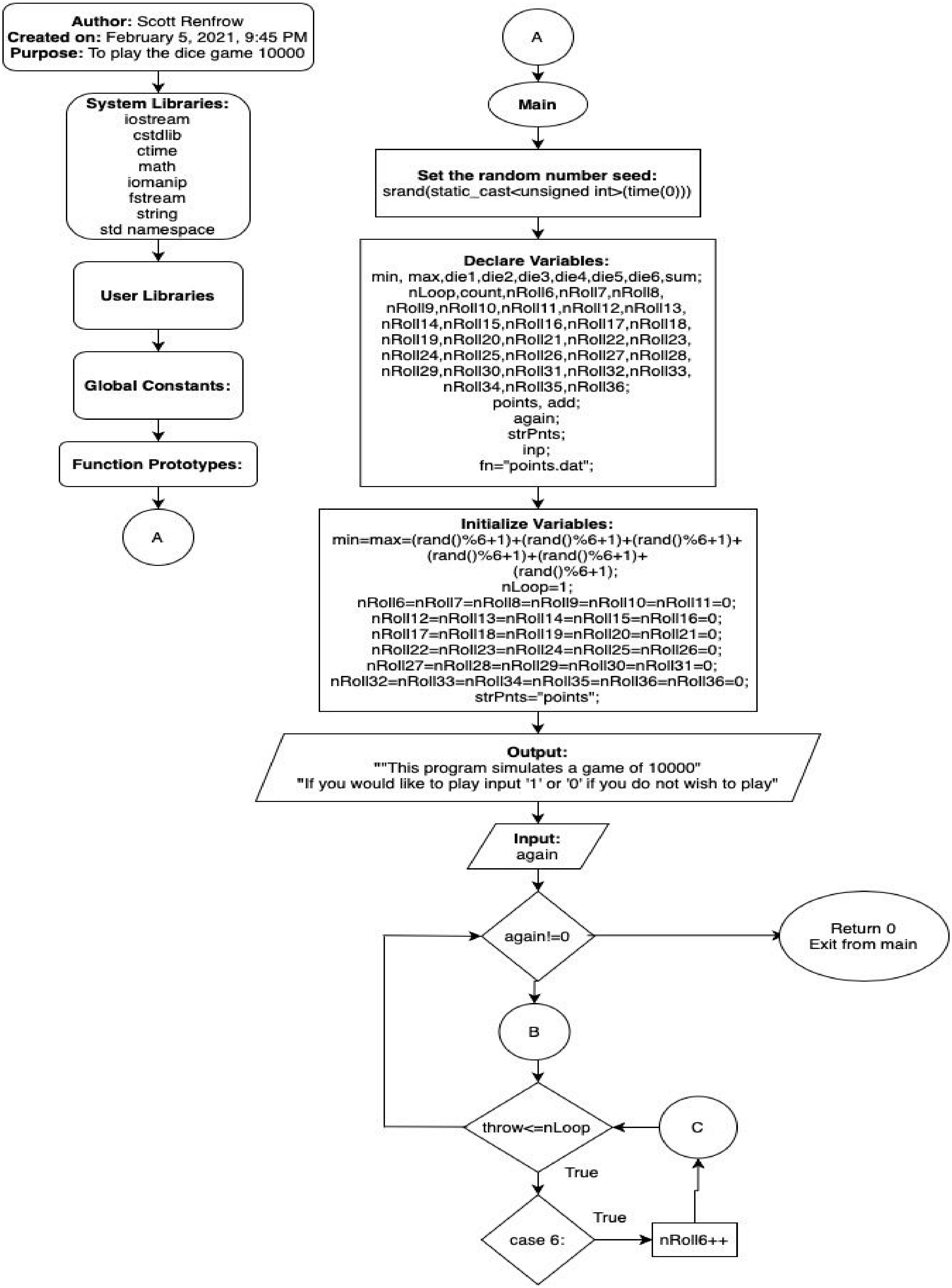
three 1’s are rolled. When three 1’s are rolled the points for that roll are 1000.

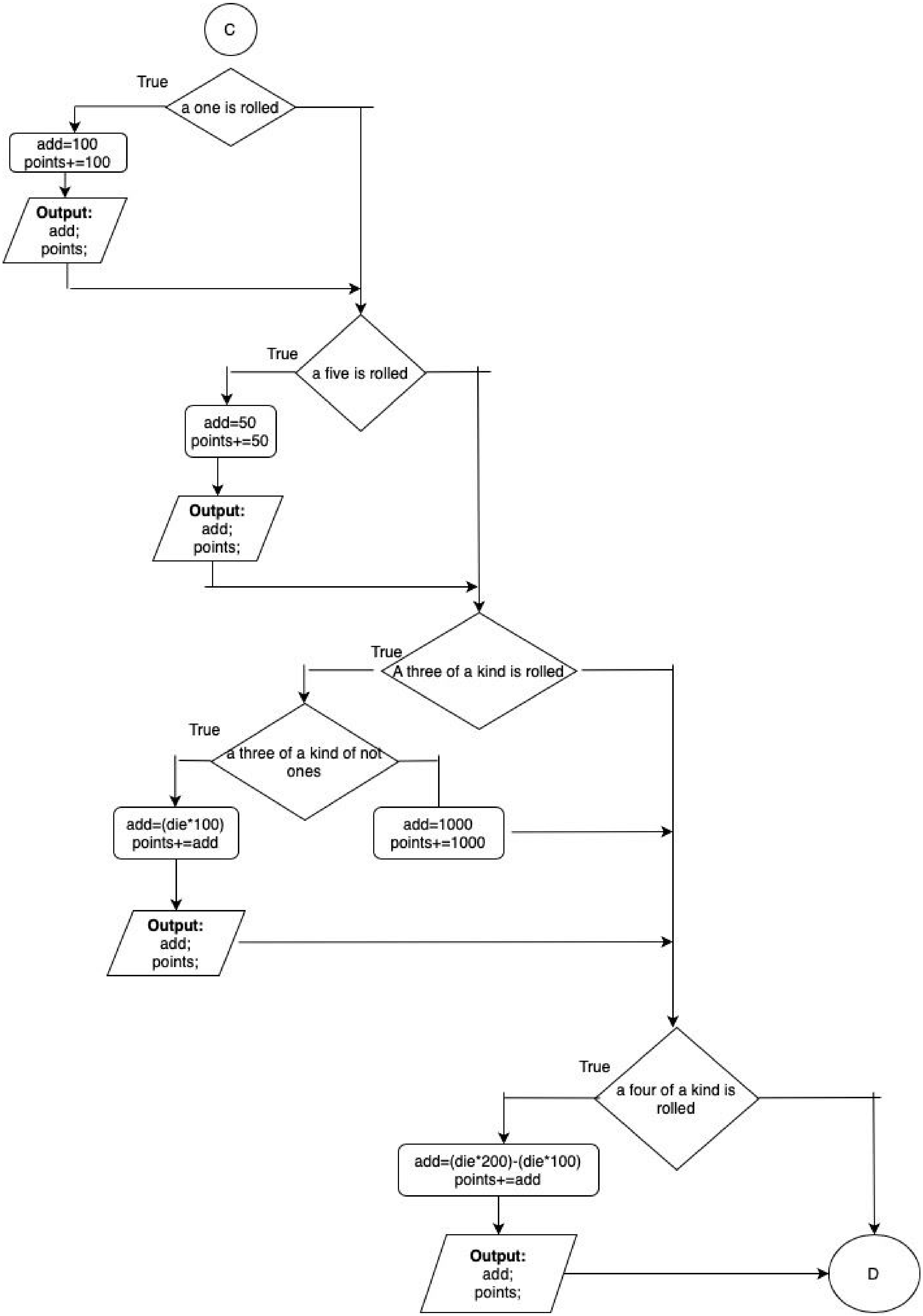
4. When a 4 of a kind is rolled, the number on the die is multiplied by 200.
5. If a 5 of a kind is rolled, it is multiplied by 400.
6. The fastest way to gain points is rolling a 6 of a kind. If this combination is rolled, the points are equal to the number rolled times 800.
7. There are also some other ways to gain points in this game however, I am not able to implement these scores into my code until project 2.

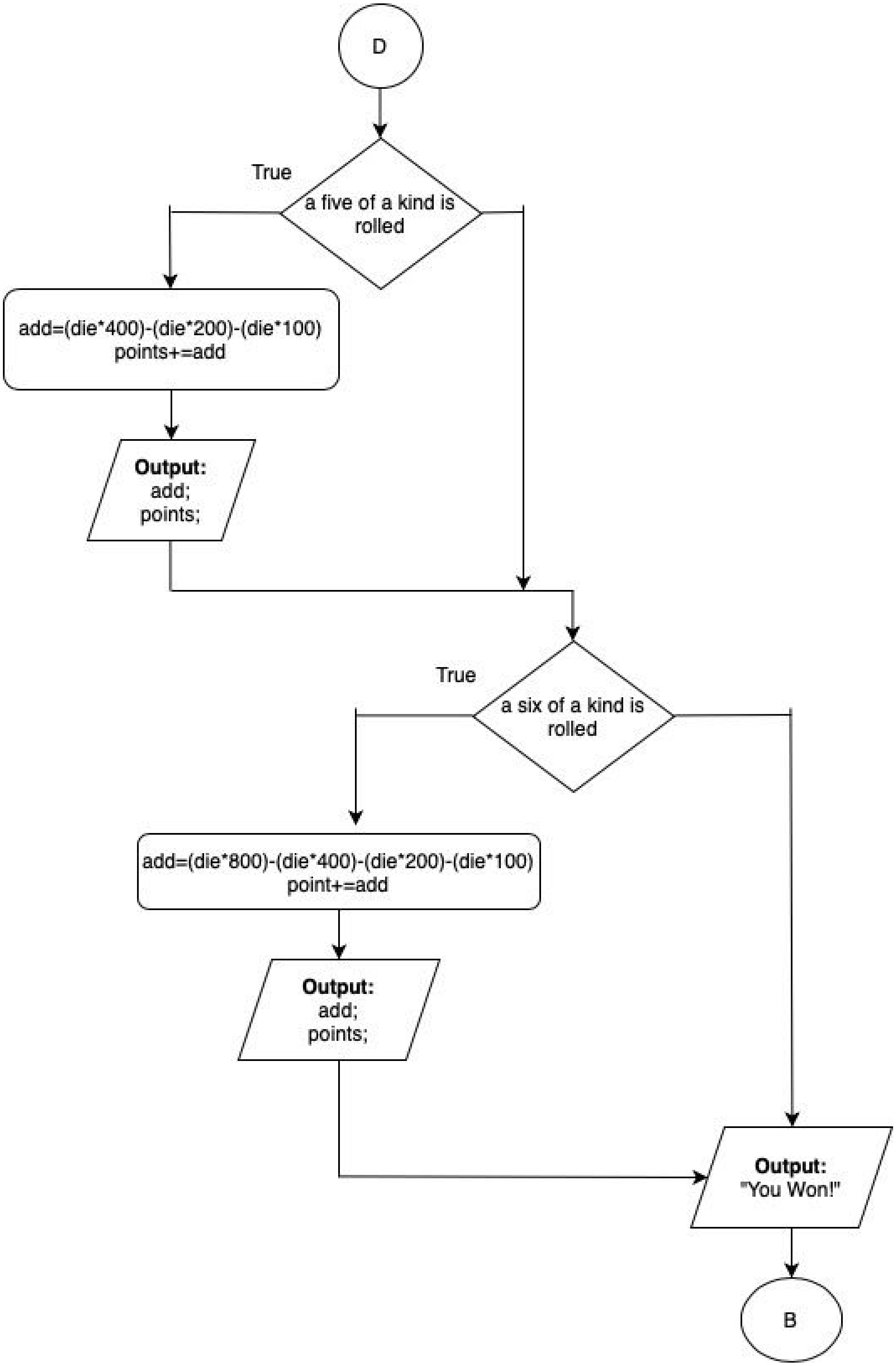
Summary:

Project Size: 700+ Lines
Number of Variables: 56
Number of Methods: 7

This project was a great experience and helped me increase my coding skills greatly. While coding this project, I ran into many problems most of which I overcame. There were a few problems I was not able to overcome however. I hope that I will be able to solve these problems in the second version of this project. This project took me about one and a half weeks to program.


Author: Scott Renfrow
Created on: February 5, 2021, 9:45 PM
Purpose: To play the dice game 10000
System Libraries:
iostream
cstdlib
ctime
math
iomanip
fstream
string
std namespace
User Libraries
Global Constants:
Function Prototypes:
A
A
Main
Set the random number seed:
srand(static_cast<unsigned int>(time(0)))
Declare Variables:
min, max,die1,die2,die3,die4,die5,die6,sum;
nLoop,count,nRoll6,nRoll7,nRoll8,
nRoll9,nRoll10,nRoll11,nRoll12,nRoll13,
nRoll14,nRoll15,nRoll16,nRoll17,nRoll18,
nRoll19,nRoll20,nRoll21,nRoll22,nRoll23,
nRoll24,nRoll25,nRoll26,nRoll27,nRoll28,
nRoll29,nRoll30,nRoll31,nRoll32,nRoll33,
nRoll34,nRoll35,nRoll36;
points, add;
again;
strPnts;
inp;
fn="points.dat";
Initialize Variables:
min=max=(rand()%6+1)+(rand()%6+1)+(rand()%6+1)+
(rand()%6+1)+(rand()%6+1)+
(rand()%6+1);
nLoop=1;
nRoll6=nRoll7=nRoll8=nRoll9=nRoll10=nRoll11=0;
nRoll12=nRoll13=nRoll14=nRoll15=nRoll16=0;
nRoll17=nRoll18=nRoll19=nRoll20=nRoll21=0;
nRoll22=nRoll23=nRoll24=nRoll25=nRoll26=0;
nRoll27=nRoll28=nRoll29=nRoll30=nRoll31=0;
nRoll32=nRoll33=nRoll34=nRoll35=nRoll36=nRoll36=0;
strPnts="points";
Output:
""This program simulates a game of 10000"
"If you would like to play input '1' or '0' if you do not wish to play"
Input:
again
again!=0
Return 0
Exit from main
B
throw<=nLoop
C
True
True
case 6:
nRoll6++


C
True
a one is rolled
add=100
points+=100
Output:
add;
points;
True
a five is rolled
add=50
points+=50
Output:
add;
points;
True
A three of a kind is rolled
True
a three of a kind of not ones
add=(die*100)
points+=add
add=1000
points+=1000
Output:
add;
points;
True
a four of a kind is rolled
add=(die*200)-(die*100)
points+=add
Output:
add;
points;
D


D
True
a five of a kind is rolled
add=(die*400)-(die*200)-(die*100)
points+=add
Output:
add;
points;
True
a six of a kind is rolled
add=(die*800)-(die*400)-(die*200)-(die*100)
point+=add
Output:
add;
points;
Output:
"You Won!"
B

Pseudo Code

```
Set the number of rolls equal to 0;
Output the menu and whether the user would like to play
Accept the input;

While the user would like to play;
    Do while the points are less than 10000;
        Randomly roll the die;
        Find the min and  max;
        Output the frequency of all values rolled;
        If a 1 is thrown
            add=100;
            points+=100;
        If a 5 is thrown;
            add=50;
            points+=50;
        If a 3 of a kind is rolled;
            If the 3 of a kind is all ones;
                add=1000;
                points+=1000;
            Else;
                add=(die*100);
                points+=add;
        If a 4 of a kind is rolled;
            add=(die*200)-(die*100);
            points+=add;
        If a 5 of a kind is rolled;
            add=(die*400)-(die*200)-(die*100);
            points+=add;
        If a 6 of a kind is rolled;
            add=(die*800)-(die*400)-(die*200)-(die*100);
            points+=add;

    "You Won!";
    "Would you like to play again?";
    Accept the input;
```

Major Variables

Type	Name	Description
Unsigned Char	min	The minimum value rolled
	max	The Maximum value rolled
	die1	The value of the first die
	die2	The value of the second die
	die3	The value of the third die
	die4	The value of the fourth die
	die5	The value of the fifth die
	die6	The value of the sixth die
	die	The dice value for the output file
Unsigned int	nLoop	The number of loops
	count	Counting sequence for the for loop
	nRoll6	The number of times a six was rolled
	nRoll7	The number of times a seven was rolled
	nRoll8	The number of times an eight was rolled
	nRoll9	The number of times a nine was rolled
	nRoll10	The number of times a ten was rolled
	nRoll11	The number of times an eleven was rolled

	nRoll12	The number of times a twelve was rolled
	nRoll13	The number of times a thirteen was rolled
	nRoll14	The number of times a fourteen was rolled
	nRoll15	The number of times a fifteen was rolled
	nRoll16	The number of times a sixteen was rolled
	nRoll17	The number of times a seventeen was rolled
	nRoll18	The number of times an eighteen was rolled
	nRoll19	The number of times a sninteen was rolled
	nRoll20	The number of times a twenty was rolled
	nRoll21	The number of times a twenty-one was rolled
	nRoll22	The number of times a twenty-two was rolled
	nRoll23	The number of times a twenty-three was rolled
	nRoll24	The number of times a twenty-four was rolled
	nRoll25	The number of times a twenty-five was rolled
	nRoll26	The number of times a twenty-six was rolled
	nRoll27	The number of times a twenty-seven was rolled
	nRoll28	The number of times a

		twenty-eight was rolled
	nRoll29	The number of times a twenty-nine was rolled
	nRoll30	The number of times a thirty was rolled
	nRoll31	The number of times a thirty-one was rolled
	nRoll32	The number of times a thirty-two was rolled
	nRoll33	The number of times a thirty-three was rolled
	nRoll34	The number of times a thirty-four was rolled
	nRoll35	The number of times a thirty-five was rolled
	nRoll36	The number of times a thirty-six was rolled
float	points	The accumulated number of points
	add	The added number of points for just one round
bool	again	Allows the user to decide if they would like to play again
string	strPnts	Displays a string value
fstream	inp	Reads in a file and outputs the value in that file
string	fn	The name of the read in file
Const int	NDICE=6	The number of dice
string	number1	The rolled number

	number2	
	number3	
	number4	
	number5	
	number6	
fstream	out	Reads out a file
string	fileNm	The file name

Checkoff Sheet

Chapter	Section	Topic	Where Line #'s	Pts	Notes
2	2	cout	153		
	3	libraries	11-17	8	iostream,iomanip,cmath,cstdlib,fstream,string,ctime
	4	variables/literals	33-64		No variables in global area, failed project!
	5	Identifiers	33-64		
	6	Integers	43-49	3	
	7	Characters	33-42	3	
	8	Strings	53	3	
	9	Floats No Doubles	50-51	3	Using doubles will fail the project, floats OK!
	10	Bools	52	4	
	11	Sizeof *****			
	12	Variables 7 characters or less	33-64		All variables <=7 characters
	13	Scope ***** No Global Variables			
	14	Arithmetic operators	79,174		
	15	Comments 20%+	106-112	5	Model as pseudo code
	16	Named constants	56		All local, only Conversions/Physics/Math in Global Area
	17	Programming Style ***** Emulate			Emulate style in book/in repository
3	1	cin	94		

	2	Math Expression	194		
	3	Mixing data types ****			
	4	Overflow/Underflow ****			
	5	Type Casting	720	4	
	6	Multiple assignment *****			
	7	Formatting output	176	4	
	8	Strings	53,163	3	
	9	Math Library	15	4	All libraries included have to be used
	10	Hand tracing ******			
4	1	Relational Operators	105		
	2	if	158-199	4	Independent if
	4	if-else	202-243	4	
	5	Nesting	246-264	4	
	6	If-else-if	265-282	4	
	7	Flags *****			
	8	Logical Operators	304	4	
	11	Validating user input	98	4	
	13	Conditional operator	115-116	4	
	14	Switch	119-154	4	
5	1	Increment/Decrement	121-151	4	
	2	While	98-760	4	
	5	Do-while	103-716	4	
	6	For loop	105-714	4	
	11	Files input/output both	68-76,99-101	8	
	12	No breaks in loops ******			Failed project if included
****** Not	required	To show	Total	100	

Program

```
/*
 *
 * File:   main.cpp
 * Author: Scott Renfrow
 * Created on February 3, 2021, 9:02 PM
 * Purpose:  Start making the game while also displaying the frequencies
 *          and using an input and output file
 */

//System Libraries
#include <iostream>  //I/O Library
#include <cstdlib>   //Random Number Generator
#include <ctime>     //Time to set the seed
#include <iomanip>   //Allows  for organization
#include <cmath>     //Math Library
#include <string>    //String Library
#include <fstream>   //File I/O
using namespace std;

//User Libraries

//Global Constants
//Math, Science, Universal, Conversions, High Dimensioned Arrays

//Function Prototypes

//Execution Begins Here
int main(int argc, char** argv) {
   //Initialize the Random Number Seed
   srand(static_cast<unsigned int>(time(0)));

   //Declare Variables
   unsigned char min, //The minimum value
              max, //The maximum value
              die1, //The value of die 1
              die2, //The value of die 2
              die3, //The value of die 3
```

```
         die4, //The value of die 4
         die5, //The value of die 5
         die6, //The value of die 6
         sum; //The sum of all 6 die
unsigned int nLoop, //The number of Loops
        count,
        nRoll6,nRoll7,nRoll8,nRoll9,nRoll10,nRoll11,nRoll12,nRoll13,
        nRoll14,nRoll15,nRoll16,nRoll17,nRoll18,nRoll19,nRoll20,
        nRoll21,nRoll22,nRoll23,nRoll24,nRoll25,nRoll26,nRoll27,
        nRoll28,nRoll29,nRoll30,nRoll31,nRoll32,nRoll33,nRoll34,
        nRoll35,nRoll36; //The frequencies of rolling each sum
float     points, //The accumulated number of points
        add;   //The added number of points for just one round
bool      again; //Allows the user to decide if they would like to play
string    strPnts;
fstream    inp;
string    fn="points.dat";
const int    NDICE=6; //The number of dice
string     number1="1",//The rolled number
         number2="2",
         number3="3",
         number4="4",
         number5="5",
         number6="6";
fstream     out;
string     fileNm;



//Initialize file parameters
fileNm="die.dat";
out.open(fileNm,ios::out);

cout<<right;
for(unsigned char die=0;die<NDICE;die++){
   string strDie;
   strDie=rand()%6+1;
}
```

```
//Initialize Variables

min=max=(rand()%6+1)+(rand()%6+1)+(rand()%6+1)+(rand()%6+1)+(rand()%6+1)+
      (rand()%6+1);
nLoop=1;
nRoll6=nRoll7=nRoll8=nRoll9=nRoll10=nRoll11=0;
nRoll12=nRoll13=nRoll14=nRoll15=nRoll16=0;
nRoll17=nRoll18=nRoll19=nRoll20=nRoll21=0;
nRoll22=nRoll23=nRoll24=nRoll25=nRoll26=0;
nRoll27=nRoll28=nRoll29=nRoll30=nRoll31=0;
nRoll32=nRoll33=nRoll34=nRoll35=nRoll36=nRoll36=0;
strPnts="points";

//Output the Menu
cout<<"This program simulates a game of 10000"<<endl;
cout<<"If you would like to play input '1' or '0' if you do";
cout<<" not wish to play."<<endl<<endl;
cin>>again;

   //Map Inputs to Outputs -> Process
   //Continue Looping until the user would like to stop
while(again!=0){
   inp.open(fn,ios::in);
   points=0;
   inp>>points; //Read in the number of points at the start of each round
   count=0;
   do{

      for(int thrw=1;thrw<=nLoop;thrw++){
         die1=(rand()%6+1); //The value for die 1
         die2=(rand()%6+1); //The value for die 2
         die3=(rand()%6+1); //The value for die 3
         die4=(rand()%6+1); //The value for die 4
         die5=(rand()%6+1); //The value for die 5
         die6=(rand()%6+1); //The value for die 6
         sum=die1+die2+die3+die4+die5+die6; //The sum of all the sides

         //Determine the minimum and maximum values
         min>sum? min=sum: min=min;
         max<sum? max=sum: max=max;
```

```
    //Display the frequencies of the numbers thrown
    switch(sum)
    {
        case 6: nRoll6++;break;
        case 7: nRoll7++;break;
        case 8: nRoll8++;break;
        case 9: nRoll9++;break;
        case 10: nRoll10++;break;
        case 11: nRoll11++;break;
        case 12: nRoll12++;break;
        case 13: nRoll13++;break;
        case 14: nRoll14++;break;
        case 15: nRoll15++;break;
        case 16: nRoll16++;break;
        case 17: nRoll17++;break;
        case 18: nRoll18++;break;
        case 19: nRoll19++;break;
        case 20: nRoll20++;break;
        case 21: nRoll21++;break;
        case 22: nRoll22++;break;
        case 23: nRoll23++;break;
        case 24: nRoll24++;break;
        case 25: nRoll25++;break;
        case 26: nRoll26++;break;
        case 27: nRoll27++;break;
        case 28: nRoll28++;break;
        case 29: nRoll29++;break;
        case 30: nRoll30++;break;
        case 31: nRoll31++;break;
        case 32: nRoll32++;break;
        case 33: nRoll33++;break;
        case 34: nRoll34++;break;
        case 35: nRoll35++;break;
        case 36: nRoll36++;break;
        default:
            cout<<"You Don't Know What is Happening"<<endl;
    }
    //Display Inputs/Outputs
cout<<"The Maximum Value Thrown = "<<static_cast<int>(max)<<endl;
```

```
cout<<"The Minimum Value Thrown = "<<static_cast<int>(min)<<endl;
cout<<"The Frequency of Rolling a 6 = "<<nRoll6<<endl;
cout<<"The Frequency of Rolling a 7 = "<<nRoll7<<endl;
cout<<"The Frequency of Rolling a 8 = "<<nRoll8<<endl;
cout<<"The Frequency of Rolling a 9 = "<<nRoll9<<endl;
cout<<"The Frequency of Rolling a 10 = "<<nRoll10<<endl;
cout<<"The Frequency of Rolling a 11 = "<<nRoll11<<endl;
cout<<"The Frequency of Rolling a 12 = "<<nRoll12<<endl;
cout<<"The Frequency of Rolling a 13 = "<<nRoll13<<endl;
cout<<"The Frequency of Rolling a 14 = "<<nRoll14<<endl;
cout<<"The Frequency of Rolling a 15 = "<<nRoll15<<endl;
cout<<"The Frequency of Rolling a 16 = "<<nRoll16<<endl;
cout<<"The Frequency of Rolling a 17 = "<<nRoll17<<endl;
cout<<"The Frequency of Rolling a 18 = "<<nRoll18<<endl;
cout<<"The Frequency of Rolling a 19 = "<<nRoll19<<endl;
cout<<"The Frequency of Rolling a 20 = "<<nRoll20<<endl;
cout<<"The Frequency of Rolling a 21 = "<<nRoll21<<endl;
cout<<"The Frequency of Rolling a 22 = "<<nRoll22<<endl;
cout<<"The Frequency of Rolling a 23 = "<<nRoll23<<endl;
cout<<"The Frequency of Rolling a 24 = "<<nRoll24<<endl;
cout<<"The Frequency of Rolling a 25 = "<<nRoll25<<endl;
cout<<"The Frequency of Rolling a 26 = "<<nRoll26<<endl;
cout<<"The Frequency of Rolling a 27 = "<<nRoll27<<endl;
cout<<"The Frequency of Rolling a 28 = "<<nRoll28<<endl;
cout<<"The Frequency of Rolling a 29 = "<<nRoll29<<endl;
cout<<"The Frequency of Rolling a 30 = "<<nRoll30<<endl;
cout<<"The Frequency of Rolling a 31 = "<<nRoll31<<endl;
cout<<"The Frequency of Rolling a 32 = "<<nRoll32<<endl;
cout<<"The Frequency of Rolling a 33 = "<<nRoll33<<endl;
cout<<"The Frequency of Rolling a 34 = "<<nRoll34<<endl;
cout<<"The Frequency of Rolling a 35 = "<<nRoll35<<endl;
cout<<"The Frequency of Rolling a 36 = "<<nRoll36<<endl;

    //Start adding points
    //Points added if a one is rolled
    if (die1==1){
        add=pow(10,2);
        points+=100.0f;
        cout<<"Added number of points from rolling a one =";
        cout<<setw(4)<<add<<"              Total number of ";
        cout<<strPnts<<" = "<<points<<endl;
    }
    if (die2==1){
        add=pow(10,2);
        points+=100.0f;
        cout<<"Added number of points from rolling a one =";
        cout<<setw(4)<<add<<"            Total number of ";
        cout<<strPnts<<" = "<<points<<endl;
    }
    if (die3==1){
        add=pow(10,2);
        points+=100.0f;
        cout<<"Added number of points from rolling a one =";
        cout<<setw(4)<<add<<"            Total number of ";
        cout<<strPnts<<" = "<<points<<endl;
    }
    if (die4==1){
        add=pow(10,2);
        points+=100.0f;
        cout<<"Added number of points from rolling a one =";
        cout<<setw(4)<<add<<"            Total number of ";
        cout<<strPnts<<" = "<<points<<endl;
    }
    if (die5==1){
        add=pow(10,2);
        points+=100.0f;
        cout<<"Added number of points from rolling a one =";
        cout<<setw(4)<<add<<"            Total number of ";
        cout<<strPnts<<" = "<<points<<endl;
    }
    if (die6==1){
        add=pow(10,2);
        points+=100.0f;
        cout<<"Added number of points from rolling a one =";
        cout<<setw(4)<<add<<"            Total number of ";
        cout<<strPnts<<" = "<<points<<endl;
    }

    //Points added if a 5 is rolled
    else if (die1==5){
       add=50.0f;
       points+=50.0f;
       cout<<"Added number of points from rolling a five = ";
       cout<<add<<"              Total number of ";
       cout<<"points = "<<points<<endl;
    }
    else if (die2==5){
       add=50.0f;
       points+=50.0f;
       cout<<"Added number of points from rolling a five = ";
       cout<<add<<"              Total number of ";
       cout<<"points = "<<points<<endl;
    }
    else if (die3==5){
       add=50.0f;
       points+=50.0f;
       cout<<"Added number of points from rolling a five = ";
       cout<<add<<"              Total number of ";
       cout<<"points = "<<points<<endl;
    }
    else if (die4==5){
       add=50.0f;
       points+=50.0f;
       cout<<"Added number of points from rolling a five = ";
       cout<<add<<"              Total number of ";
       cout<<"points = "<<points<<endl;
    }
    else if (die5==5){
       add=50.0f;
       points+=50.0f;
       cout<<"Added number of points from rolling a five = ";
       cout<<add<<"              Total number of ";
       cout<<"points = "<<points<<endl;
    }
    else if (die6==5){
       add=50.0f;
       points+=50.0f;
       cout<<"Added number of points from rolling a five = ";
       cout<<add<<"              Total number of ";
       cout<<"points = "<<points<<endl;
```

```
 }

 //Points added if a 3 of a kind is rolled
 if (die1==die2 && die2==die3)
 {
    if (die1==1)
    {
       add=1000.0f;
       points+=1000.0f;
       cout<<"Added number of points from rolling three ones";
       cout<<" = "<<add<<"         Total number of ";
       cout<<"points = "<<points<<endl;
    }
    else{
    add=(die1*100.0f);
    points=points+(die1*100.0f);
    cout<<"Added number of points from rolling a three";
    cout<<" of a kind";
       cout<<" = "<<add<<"   Total number of ";
       cout<<"points = "<<points<<endl;
    }
 }
if (die1==die2 && die2==die4)
 {
   if (die1==1)
    {
       add=1000.0f;
       points+=1000.0f;
       cout<<"Added number of points from rolling three ones";
       cout<<" = "<<add<<"         Total number of ";
       cout<<"points = "<<points<<endl;
    }
    else{
      add=(die1*100.0f);
    points=points+(die1*100.0f);
    cout<<"Added number of points from rolling a";
    cout<<" three of a kind";
       cout<<" = "<<add<<"   Total number of ";
       cout<<"points = "<<points<<endl;
    }
```

```
}
if (die1==die2 && die2==die5)
{
    if (die1==1)
    {
        add=1000.0f;
        points+=1000.0f;
        cout<<"Added number of points from rolling three ones";
        cout<<" = "<<add<<"         Total number of ";
        cout<<"points = "<<points<<endl;
    }
    else{
        add=(die1*100.0f);
    points=points+(die1*100.0f);
    cout<<"Added number of points from rolling a three of a";
        cout<<" kind";
        cout<<" = "<<add<<"   Total number of ";
        cout<<"points = "<<points<<endl;
    }
}
if (die1==die2 && die2==6)
{
    if (die1==1)
    {
        add=1000.0f;
        points+=1000.0f;
        cout<<"Added number of points from rolling three ones";
        cout<<" = "<<add<<"         Total number of ";
        cout<<"points = "<<points<<endl;
    }
    else{
        add=(die1*100.0f);
    points=points+(die1*100.0f);
    cout<<"Added number of points from rolling a three";
        cout<<" of a kind";
        cout<<" = "<<add<<"   Total number of ";
        cout<<"points = "<<points<<endl;
    }
}
```

```
if (die1==die3 && die3==die4)
{
   if (die1==1)
   {
      add=1000.0f;
      points+=1000.0f;
      cout<<"Added number of points from rolling three ones";
      cout<<" = "<<add<<"         Total number of ";
      cout<<"points = "<<points<<endl;
   }
   else{
      add=(die1*100.0f);
   points=points+(die1*100.0f);
   cout<<"Added number of points from rolling a three";
      cout<<" of a kind";
      cout<<" = "<<add<<"   Total number of ";
      cout<<"points = "<<points<<endl;
   }
}
if (die1==die3 && die3==die5)
{
   if (die1==1)
   {
      add=1000.0f;
      points+=1000.0f;
      cout<<"Added number of points from rolling three ones";
      cout<<" = "<<add<<"         Total number of ";
      cout<<"points = "<<points<<endl;
   }
   else{
      add=(die1*100.0f);
   points=points+(die1*100.0f);
   cout<<"Added number of points from rolling a three";
      cout<<" of a kind";
      cout<<" = "<<add<<"   Total number of ";
      cout<<"points = "<<points<<endl;
   }
}
if (die1==die3 && die3==die6)
{
   if (die1==1)
   {
      add=1000.0f;
      points+=1000.0f;
      cout<<"Added number of points from rolling three ones";
      cout<<" = "<<add<<"         Total number of ";
      cout<<"points = "<<points<<endl;
   }
   else{
   points=points+(die1*100.0f);
   cout<<"Added number of points from rolling a three";
      cout<<" of a kind";
      cout<<" = "<<add<<"   Total number of ";
      cout<<"points = "<<points<<endl;
   }
}
if (die2==die3 && die3==die4)
{
   if (die2==1)
   {
      add=1000.0f;
      points+=1000.0f;
      cout<<"Added number of points from rolling three ones";
      cout<<" = "<<add<<"         Total number of ";
      cout<<"points = "<<points<<endl;
   }
   else{
      add=(die2*100.0f);
   points=points+(die2*100.0f);
   cout<<"Added number of points from rolling a three";
      cout<<" of a kind";
      cout<<" = "<<add<<"   Total number of ";
      cout<<"points = "<<points<<endl;
   }
}
if (die2==die3 && die3==die5)
{
   if (die2==1)
   {
      add=1000.0f;
      points+=1000.0f;
      cout<<"Added number of points from rolling three ones";
      cout<<" = "<<add<<"         Total number of ";
      cout<<"points = "<<points<<endl;
   }
   else{
      add=(die2*100.0f);
   points=points+(die2*100.0f);
   cout<<"Added number of points from rolling a three";
      cout<<" of a kind";
      cout<<" = "<<add<<"   Total number of ";
      cout<<"points = "<<points<<endl;
   }
}
if (die2==die3 && die3==die6)
{
   if (die2==1)
   {
      add=1000.0f;
      points+=1000.0f;
      cout<<"Added number of points from rolling three ones";
      cout<<" = "<<add<<"         Total number of ";
      cout<<"points = "<<points<<endl;
   }
   else{
      add=(die2*100.0f);
   points=points+(die2*100.0f);
   cout<<"Added number of points from rolling a three";
      cout<<" of a kind";
      cout<<" = "<<add<<"   Total number of ";
      cout<<"points = "<<points<<endl;
   }
}
if (die2==die4 && die4==die5)
{
   if (die2==1)
   {
      add=1000.0f;
      points+=1000.0f;
      cout<<"Added number of points from rolling three ones";
```

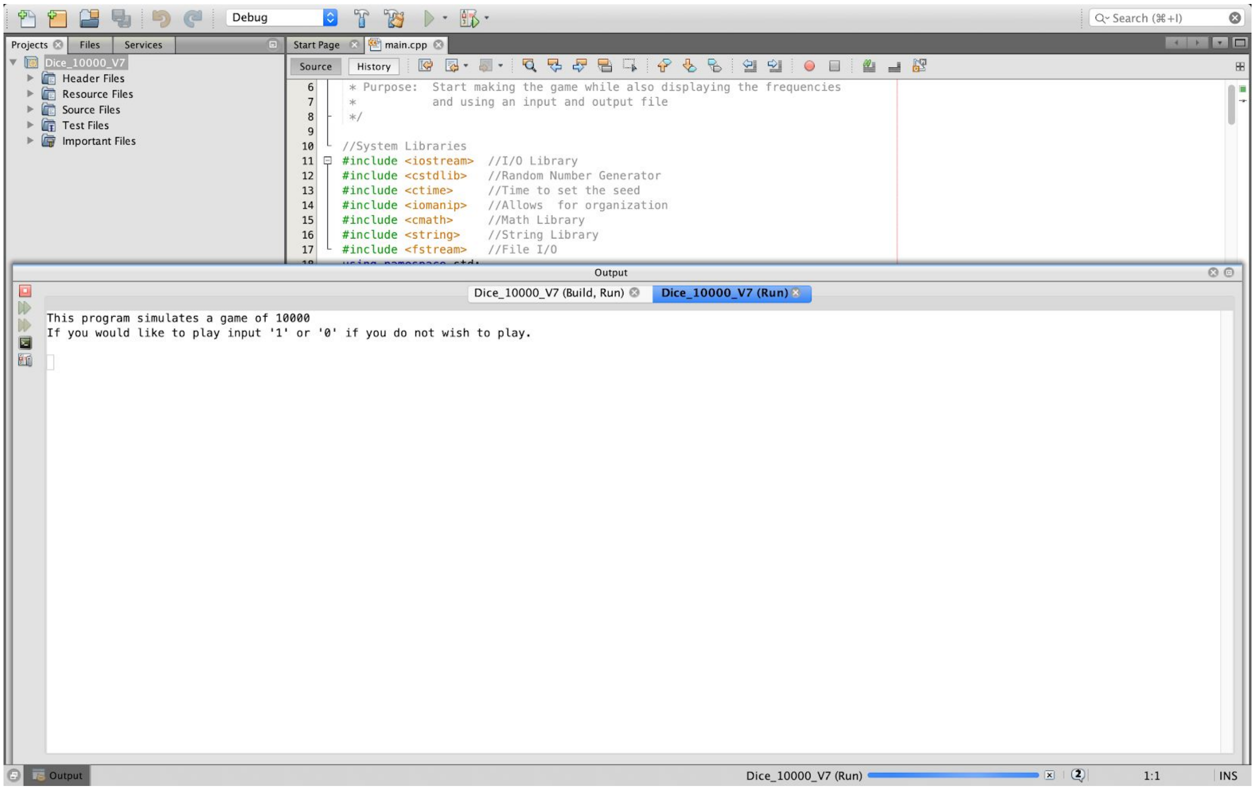
```
        cout<<" = "<<add<<"          Total number of ";
        cout<<"points = "<<points<<endl;
    }
    else{
        add=(die2*100.0f);
    points=points+(die2*100.0f);
    cout<<"Added number of points from rolling a three";
        cout<<" of a kind";
        cout<<" = "<<add<<"    Total number of ";
        cout<<"points = "<<points<<endl;
    }
}
if (die2==die4 && die4==die6)
{
    if (die2==1)
    {
        add=1000.0f;
        points+=1000.0f;
        cout<<"Added number of points from rolling three ones";
        cout<<" = "<<add<<"          Total number of ";
        cout<<"points = "<<points<<endl;
    }
    else{
        add=(die2*100.0f);
    points=points+(die2*100.0f);
    cout<<"Added number of points from rolling a three";
        cout<<" of a kind";
        cout<<" = "<<add<<"    Total number of ";
        cout<<"points = "<<points<<endl;
    }
}
if (die3==die4 && die4==die5)
{
    if (die3==1)
    {
        add=1000.0f;
        points+=1000.0f;
        cout<<"Added number of points from rolling three ones";
        cout<<" = "<<add<<"          Total number of ";
        cout<<"points = "<<points<<endl;
    }
    else{
        add=(die3*100.0f);
    points=points+(die3*100.0f);
    cout<<"Added number of points from rolling a three";
        cout<<" of a kind";
        cout<<" = "<<add<<"   Total number of ";
        cout<<"points = "<<points<<endl;
    }
}
if (die3==die4 && die4==die6)
{
    if (die3==1)
    {
        add=1000.0f;
        points+=1000.0f;
        cout<<"Added number of points from rolling three ones";
        cout<<" = "<<add<<"         Total number of ";
        cout<<"points = "<<points<<endl;
    }
    else{
        add=(die3*100.0f);
    points=points+(die3*100.0f);
    cout<<"Added number of points from rolling a three";
        cout<<" of a kind";
        cout<<" = "<<add<<"   Total number of ";
        cout<<"points = "<<points<<endl;
    }
}
if (die3==die5 && die5==die6)
{
    if (die3==1)
    {
        add=1000.0f;
        points+=1000.0f;
        cout<<"Added number of points from rolling three ones";
        cout<<" = "<<add<<"         Total number of ";
        cout<<"points = "<<points<<endl;
    }
    else{
        add=(die3*100.0f);
    points=points+(die3*100.0f);
    cout<<"Added number of points from rolling a three";
        cout<<" of a kind";
        cout<<" = "<<add<<"   Total number of ";
        cout<<"points = "<<points<<endl;
    }
}

//The points added if a four of a kind is rolled
if (die1==die2 && die2==die3 && die3==die4){
    add=(die1*200.0f)-(die1*100.0f);
    points=points+(die1*200.0f)-(die1*100.0f);
    cout<<"Added number of points from rolling a four";
        cout<<" of a kind";
        cout<<" = "<<add<<"    Total number of ";
        cout<<"points = "<<points<<endl;
}
if (die1==die2 && die2==die3 && die3==die5){
    add=(die1*200.0f)-(die1*100.0f);
    points=points+(die1*200.0f)-(die1*100.0f);
    cout<<"Added number of points from rolling a four";
        cout<<" of a kind";
        cout<<" = "<<add<<"    Total number of ";
        cout<<"points = "<<points<<endl;
}
if (die1==die2 && die2==die3 && die3==die6){
    add=(die1*200.0f)-(die1*100.0f);
    points=points+(die1*200.0f)-(die1*100.0f);
    cout<<"Added number of points from rolling a four";
        cout<<" of a kind";
        cout<<" = "<<add<<"    Total number of ";
        cout<<"points = "<<points<<endl;
}
if (die1==die2 && die2==die4 && die4==die5){
    add=(die1*200.0f)-(die1*100.0f);
    points=points+(die1*200.0f)-(die1*100.0f);
    cout<<"Added number of points from rolling a four";
        cout<<" of a kind ";
        cout<<" = "<<add<<"    Total number of ";
        cout<<"points = "<<points<<endl;
}
if (die1==die2 && die2==die4 && die4==die6){
    add=(die1*200.0f)-(die1*100.0f);
    points=points+(die1*200.0f)-(die1*100.0f);
    cout<<"Added number of points from rolling a four";
        cout<<" of a kind ";
        cout<<" = "<<add<<"    Total number of ";
        cout<<"points = "<<points<<endl;
}
if (die1==die2 && die2==die5 && die5==die6){
    add=(die1*200.0f)-(die1*100.0f);
    points=points+(die1*200.0f)-(die1*100.0f);
    cout<<"Added number of points from rolling a four";
        cout<<" of a kind";
        cout<<" = "<<add<<"    Total number of ";
        cout<<"points = "<<points<<endl;
}
if (die1==die3 && die3==die4 && die4==die5){
    add=(die1*200.0f)-(die1*100.0f);
    points=points+(die1*200.0f)-(die1*100.0f);
    cout<<"Added number of points from rolling a four";
        cout<<" of a kind";
        cout<<" = "<<add<<"    Total number of ";
        cout<<"points = "<<points<<endl;
}
if (die1==die3 && die3==die4 && die4==die6){
    add=(die1*200.0f)-(die1*100.0f);
    points=points+(die1*200.0f)-(die1*100.0f);
    cout<<"Added number of points from rolling a four";
        cout<<" of a kind";
        cout<<" = "<<add<<"    Total number of ";
        cout<<"points = "<<points<<endl;
}
if (die1==die3 && die3==die5 && die5==die6){
    add=(die1*200.0f)-(die1*100.0f);
    points=points+(die1*200.0f)-(die1*100.0f);
    cout<<"Added number of points from rolling a four";
        cout<<" of a kind";
        cout<<" = "<<add<<"    Total number of ";
        cout<<"points = "<<points<<endl;
}
if (die1==die4 && die4==die5 && die5==die6){
    add=(die1*200.0f)-(die1*100.0f);
    points=points+(die1*200.0f)-(die1*100.0f);
    cout<<"Added number of points from rolling a four";
        cout<<" of a kind";
        cout<<" = "<<add<<"    Total number of ";
        cout<<"points = "<<points<<endl;
}
if (die2==die3 && die3==die4 && die4==die5){
    add=(die2*200.0f)-(die2*100.0f);
    points=points+(die2*200.0f)-(die2*100.0f);
    cout<<"Added number of points from rolling a four";
        cout<<" of a kind";
        cout<<" = "<<add<<"    Total number of ";
        cout<<"points = "<<points<<endl;
}
if (die2==die3 && die3==die4 && die4==die6){
    add=(die2*200.0f)-(die2*100.0f);
    points=points+(die2*200.0f)-(die2*100.0f);
    cout<<"Added number of points from rolling a four";
        cout<<" of a kind";
        cout<<" = "<<add<<"    Total number of ";
        cout<<"points = "<<points<<endl;
}
if (die2==die3 && die3==die5 && die5==die6){
    add=(die2*200.0f)-(die2*100.0f);
    points=points+(die2*200.0f)-(die2*100.0f);
    cout<<"Added number of points from rolling a four";
        cout<<" of a kind";
        cout<<" = "<<add<<"    Total number of ";
        cout<<"points = "<<points<<endl;
}
if (die2==die4 && die4==die5 && die5==die6){
    add=(die2*200.0f)-(die2*100.0f);
    points=points+(die2*200.0f)-(die2*100.0f);
    cout<<"Added number of points from rolling a four";
        cout<<" of a kind";
        cout<<" = "<<add<<"    Total number of ";
        cout<<"points = "<<points<<endl;
}
if (die3==die4 && die4==die5 && die5==die6){
    add=(die3*200.0f)-(die3*100.0f);
    points=points+(die3*200.0f)-(die3*100.0f);
    cout<<"Added number of points from rolling a four";
        cout<<" of a kind";
        cout<<" = "<<add<<"    Total number of ";
        cout<<"points = "<<points<<endl;
}

//The points added if a five of a kind is rolled
if (die1==die2 && die2==die3 && die3==die4 && die4==die5){
    add=(die1*400.0f)-(die1*200.0f)-(die1*100.0f);
    points=points+(die1*400.0f)-(die1*200.0f)-(die1*100.0f);
    cout<<"Added number of points from rolling a five";
        cout<<" of a kind";
        cout<<" = "<<add<<"    Total number of ";
        cout<<"points = "<<points<<endl;
}
if (die1==die2 && die2==die3 && die3==die4 && die4==die6){
    add=(die1*400.0f)-(die1*200.0f)-(die1*100.0f);
    points=points+(die1*400.0f)-(die1*200.0f)-(die1*100.0f);
    cout<<"Added number of points from rolling a five";
        cout<<" of a kind";
        cout<<" = "<<add<<"    Total number of ";
        cout<<"points = "<<points<<endl;
}
if (die1==die2 && die2==die3 && die3==die5 && die5==die6){
    add=(die1*400.0f)-(die1*200.0f)-(die1*100.0f);
    points=points+(die1*400.0f)-(die1*200.0f)-(die1*100.0f);
    cout<<"Added number of points from rolling a five";
        cout<<" of a kind";
        cout<<" = "<<add<<"    Total number of ";
        cout<<"points = "<<points<<endl;
}
if (die1==die2 && die2==die4 && die4==die5 && die5==die6){
    add=(die1*400.0f)-(die1*200.0f)-(die1*100.0f);
    points=points+(die1*400.0f)-(die1*200.0f)-(die1*100.0f);
    cout<<"Added number of points from rolling a five";
```

```
                cout<<" of a kind";
                cout<<" = "<<add<<"    Total number of ";
                cout<<"points = "<<points<<endl;
        }
        if (die1==die3 && die3==die4 && die4==die5 && die5==die6){
            add=(die1*400.0f)-(die1*200.0f)-(die1*100.0f);
            points=points+(die1*400.0f)-(die1*200.0f)-(die1*100.0f);
            cout<<"Added number of points from rolling a five";
                cout<<" of a kind";
                cout<<" = "<<add<<"    Total number of ";
                cout<<"points = "<<points<<endl;
        }
        if (die2==die3 && die3==die4 && die4==die5 && die5==die6){
            add=(die2*400.0f)-(die2*200.0f)-(die2*100.0f);
            points=points+(die2*400.0f)-(die2*200.0f)-(die2*100.0f);
            cout<<"Added number of points from rolling a five";
                cout<<" of a kind";
                cout<<" = "<<add<<"    Total number of ";
                cout<<"points = "<<points<<endl;
        }

        //The points added if a six of a kind is rolled
        if (die1==die2&&die2==die3&&die3==die4&&die4==die5&&die5==die6){
            add=(die1*800.0f)-(die1*400.0f)-(die1*200.0f)-(die1*100.0f);
            points=points+(die1*800.0f)-(die1*400.0f)-(die1*200.0f);
            points=points-(die1*200.0f);
            cout<<"Added number of points from rolling a six of a kind";
                cout<<" = "<<add<<"    Total number of ";
                cout<<"points = "<<points<<endl;
        }
    count++;
    }

}while (points<10000); //Continue this loop until 10000 points

//Output "You Won!"
cout<<endl<<"You Won!"<<endl;

//Ask the user if they would like to play again
cout<<"If you would like to play again input '1' or '0' if you do";
```

```
        cout<<" not wish to play again."<<endl<<endl;
        cin>>again;

    }
    //Exit the Program - Cleanup
    inp.close();
    out.close();
    return 0;
}
```

Example Outputs

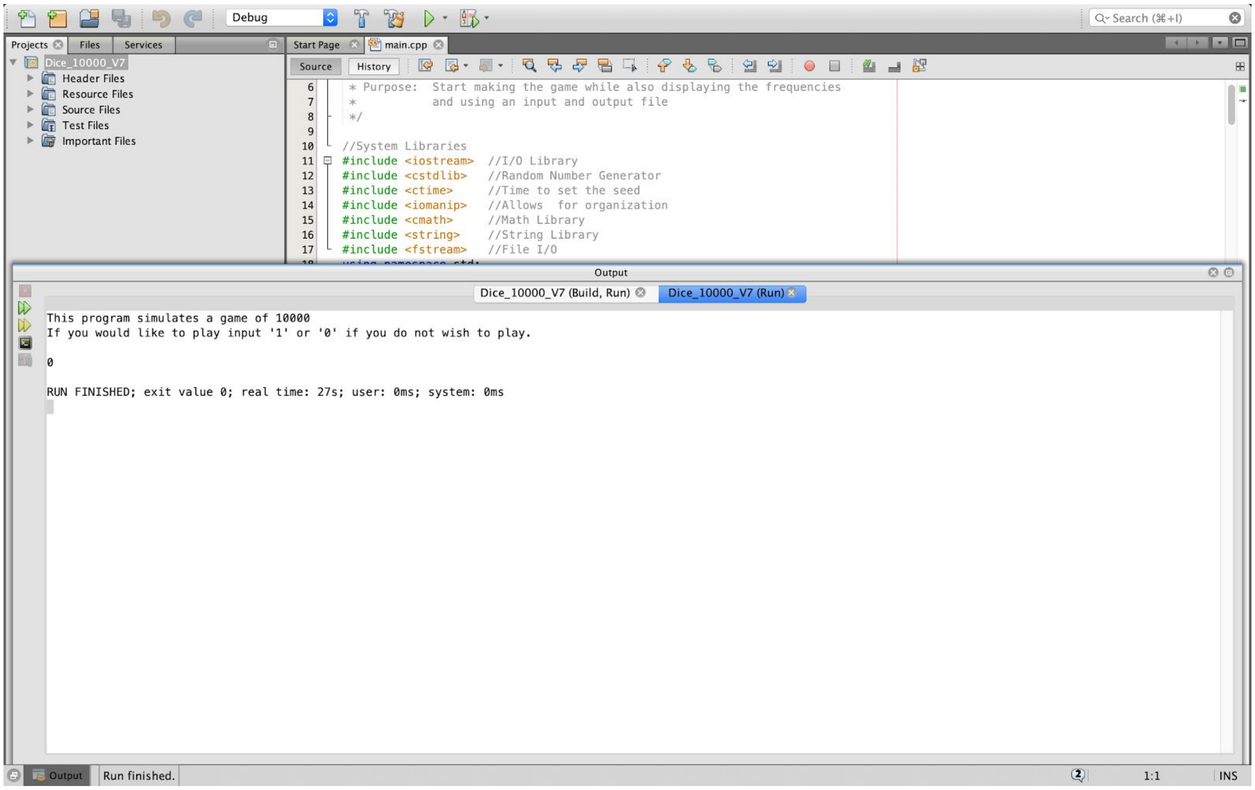
This example outputs the opening menu for the program and shows that the user can either enter “1” to play the game, or a “0” to end the program.



```
 * Purpose:  Start making the game while also displaying the frequencies
 *           and using an input and output file
 */

//System Libraries
#include <iostream>  //I/O Library
#include <cstdlib>   //Random Number Generator
#include <ctime>     //Time to set the seed
#include <iomanip>   //Allows  for organization
#include <cmath>     //Math Library
#include <string>    //String Library
#include <fstream>   //File I/O
using namespace std;
```

```
This program simulates a game of 10000
If you would like to play input '1' or '0' if you do not wish to play.
```

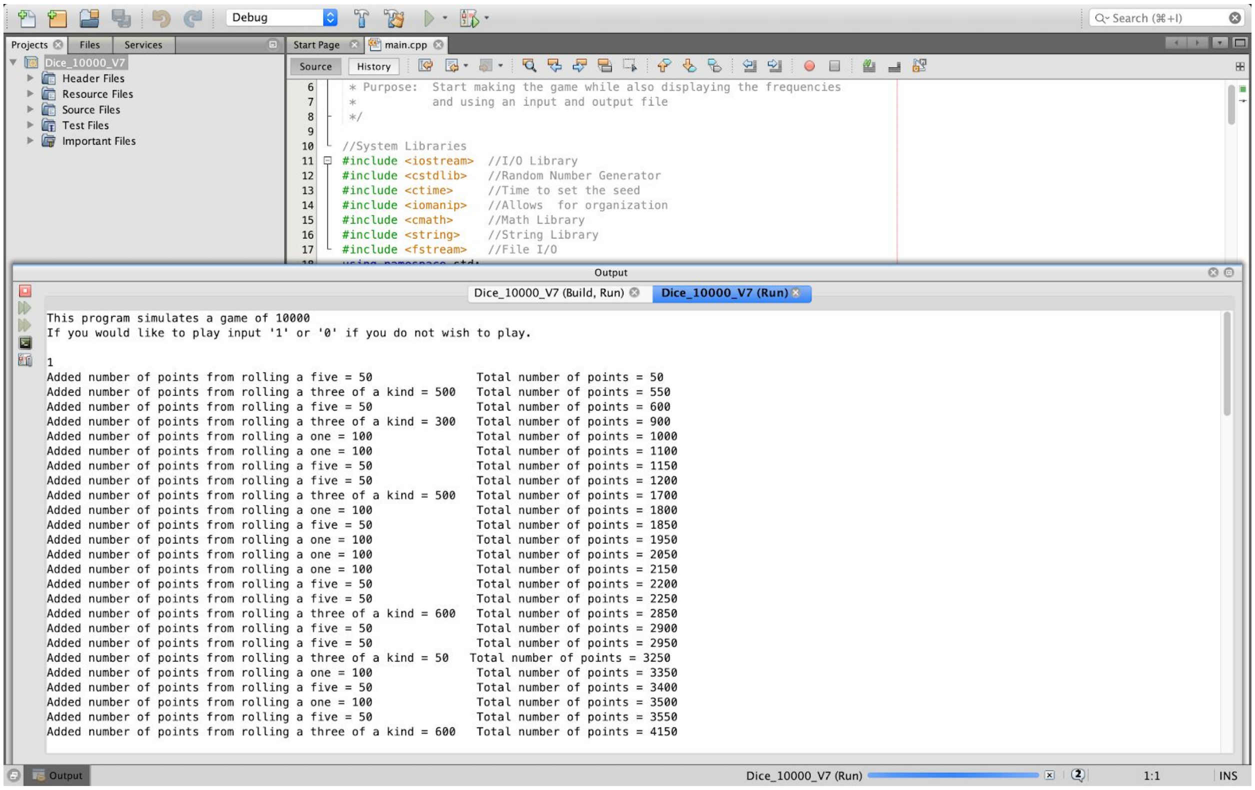
In this example, you can see that the user inputted a “0,” causing the program to end.



```
 * Purpose:  Start making the game while also displaying the frequencies
 *           and using an input and output file
 */

//System Libraries
#include <iostream>  //I/O Library
#include <cstdlib>   //Random Number Generator
#include <ctime>     //Time to set the seed
#include <iomanip>   //Allows  for organization
#include <cmath>     //Math Library
#include <string>    //String Library
#include <fstream>   //File I/O
using namespace std;
```

```
This program simulates a game of 10000
If you would like to play input '1' or '0' if you do not wish to play.

0

RUN FINISHED; exit value 0; real time: 27s; user: 0ms; system: 0ms
```

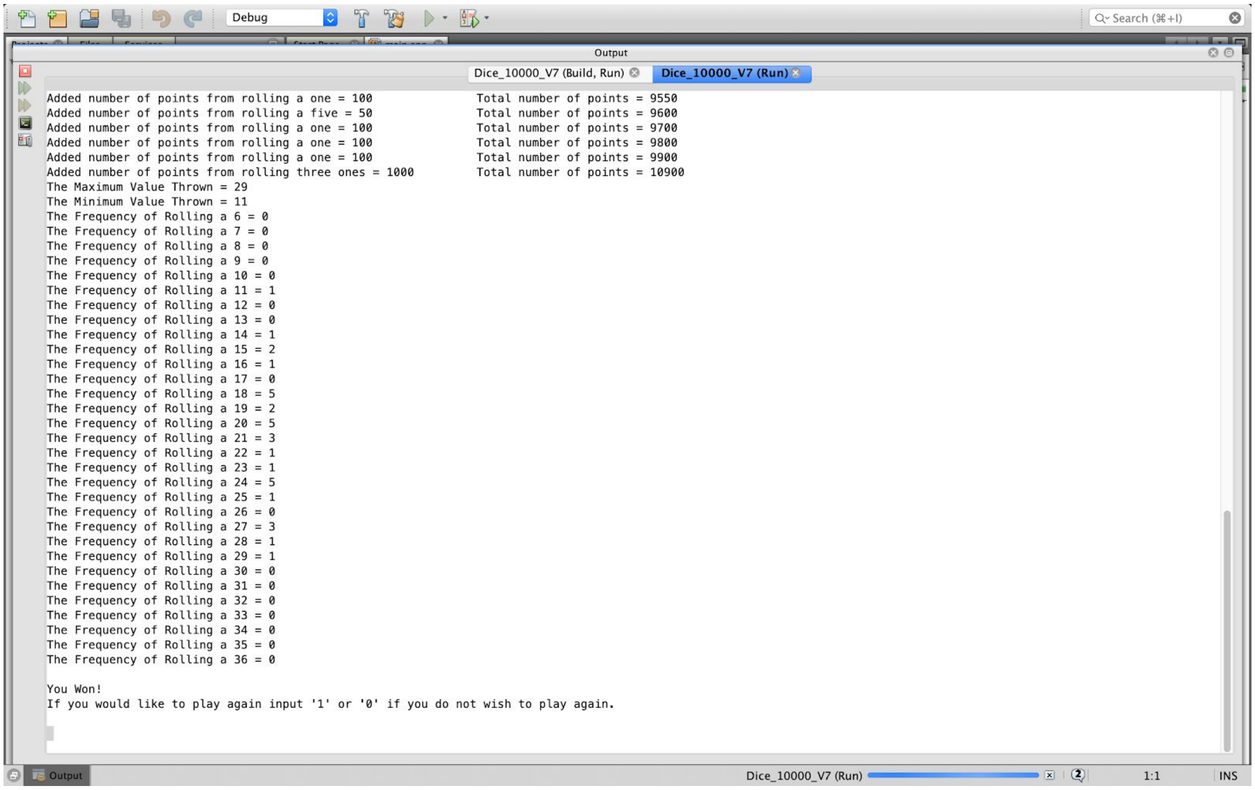
Here is an example, where the user decided to run the program by inputting a “1.” You can also see that the total number of points are slowly being added.



```
 * Purpose:  Start making the game while also displaying the frequencies
 *           and using an input and output file
 */

//System Libraries
#include <iostream>  //I/O Library
#include <cstdlib>   //Random Number Generator
#include <ctime>     //Time to set the seed
#include <iomanip>   //Allows  for organization
#include <cmath>     //Math Library
#include <string>    //String Library
#include <fstream>   //File I/O
```

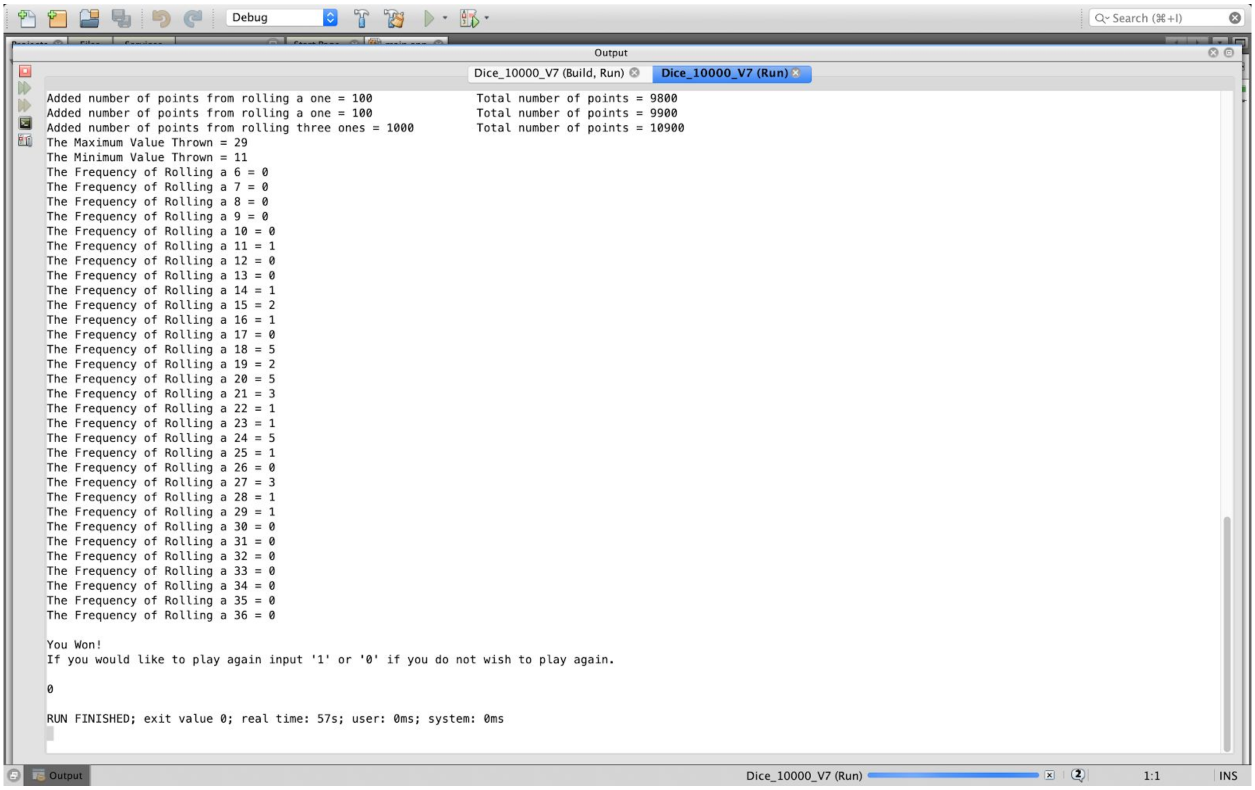
```
This program simulates a game of 10000
If you would like to play input '1' or '0' if you do not wish to play.

1
Added number of points from rolling a five = 50             Total number of points = 50
Added number of points from rolling a three of a kind = 500   Total number of points = 550
Added number of points from rolling a five = 50             Total number of points = 600
Added number of points from rolling a three of a kind = 300   Total number of points = 900
Added number of points from rolling a one = 100             Total number of points = 1000
Added number of points from rolling a one = 100             Total number of points = 1100
Added number of points from rolling a five = 50             Total number of points = 1150
Added number of points from rolling a five = 50             Total number of points = 1200
Added number of points from rolling a three of a kind = 500   Total number of points = 1700
Added number of points from rolling a one = 100             Total number of points = 1800
Added number of points from rolling a five = 50             Total number of points = 1850
Added number of points from rolling a one = 100             Total number of points = 1950
Added number of points from rolling a one = 100             Total number of points = 2050
Added number of points from rolling a one = 100             Total number of points = 2150
Added number of points from rolling a five = 50             Total number of points = 2200
Added number of points from rolling a five = 50             Total number of points = 2250
Added number of points from rolling a three of a kind = 600   Total number of points = 2850
Added number of points from rolling a five = 50             Total number of points = 2900
Added number of points from rolling a five = 50             Total number of points = 2950
Added number of points from rolling a three of a kind = 50   Total number of points = 3250
Added number of points from rolling a one = 100             Total number of points = 3350
Added number of points from rolling a five = 50             Total number of points = 3400
Added number of points from rolling a one = 100             Total number of points = 3500
Added number of points from rolling a five = 50             Total number of points = 3550
Added number of points from rolling a three of a kind = 600   Total number of points = 4150
```

In this example, you can see that the program stopped running once the points reached 10,000 because at that point the player had won. I have also caused the program to output the minimum and maximum values rolled from that round. Additionally, the frequency of each roll is displayed. From here, the user can either decide to play again by inputting a “1” or stop the program by inputting a “0”.



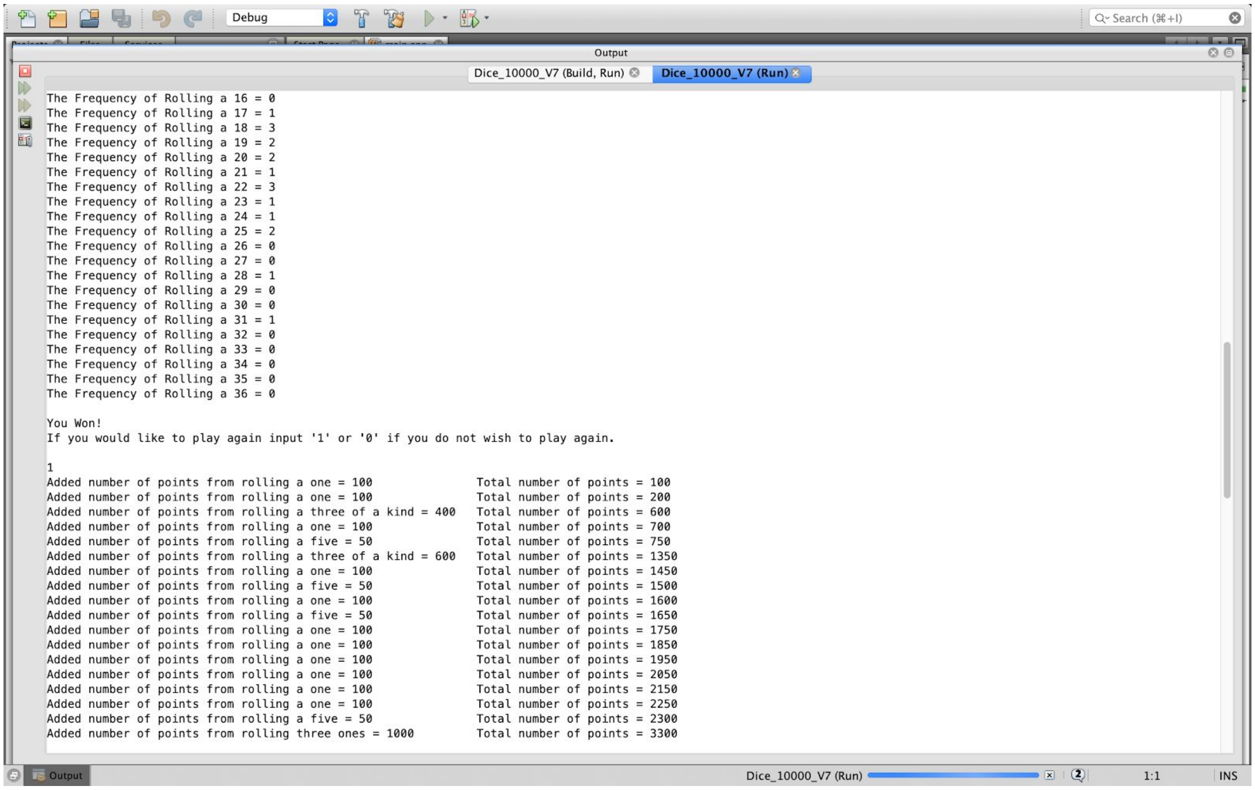
```
Added number of points from rolling a one = 100               Total number of points = 9550
Added number of points from rolling a five = 50               Total number of points = 9600
Added number of points from rolling a one = 100               Total number of points = 9700
Added number of points from rolling a one = 100               Total number of points = 9800
Added number of points from rolling a one = 100               Total number of points = 9900
Added number of points from rolling three ones = 1000         Total number of points = 10900
The Maximum Value Thrown = 29
The Minimum Value Thrown = 11
The Frequency of Rolling a 6 = 0
The Frequency of Rolling a 7 = 0
The Frequency of Rolling a 8 = 0
The Frequency of Rolling a 9 = 0
The Frequency of Rolling a 10 = 0
The Frequency of Rolling a 11 = 1
The Frequency of Rolling a 12 = 0
The Frequency of Rolling a 13 = 0
The Frequency of Rolling a 14 = 1
The Frequency of Rolling a 15 = 2
The Frequency of Rolling a 16 = 1
The Frequency of Rolling a 17 = 0
The Frequency of Rolling a 18 = 5
The Frequency of Rolling a 19 = 2
The Frequency of Rolling a 20 = 5
The Frequency of Rolling a 21 = 3
The Frequency of Rolling a 22 = 1
The Frequency of Rolling a 23 = 1
The Frequency of Rolling a 24 = 5
The Frequency of Rolling a 25 = 1
The Frequency of Rolling a 26 = 0
The Frequency of Rolling a 27 = 3
The Frequency of Rolling a 28 = 1
The Frequency of Rolling a 29 = 1
The Frequency of Rolling a 30 = 0
The Frequency of Rolling a 31 = 0
The Frequency of Rolling a 32 = 0
The Frequency of Rolling a 33 = 0
The Frequency of Rolling a 34 = 0
The Frequency of Rolling a 35 = 0
The Frequency of Rolling a 36 = 0

You Won!
If you would like to play again input '1' or '0' if you do not wish to play again.
```

Here is an example showing what happens if the user would not like to run the program again. By inputting a “0,” the program ends.



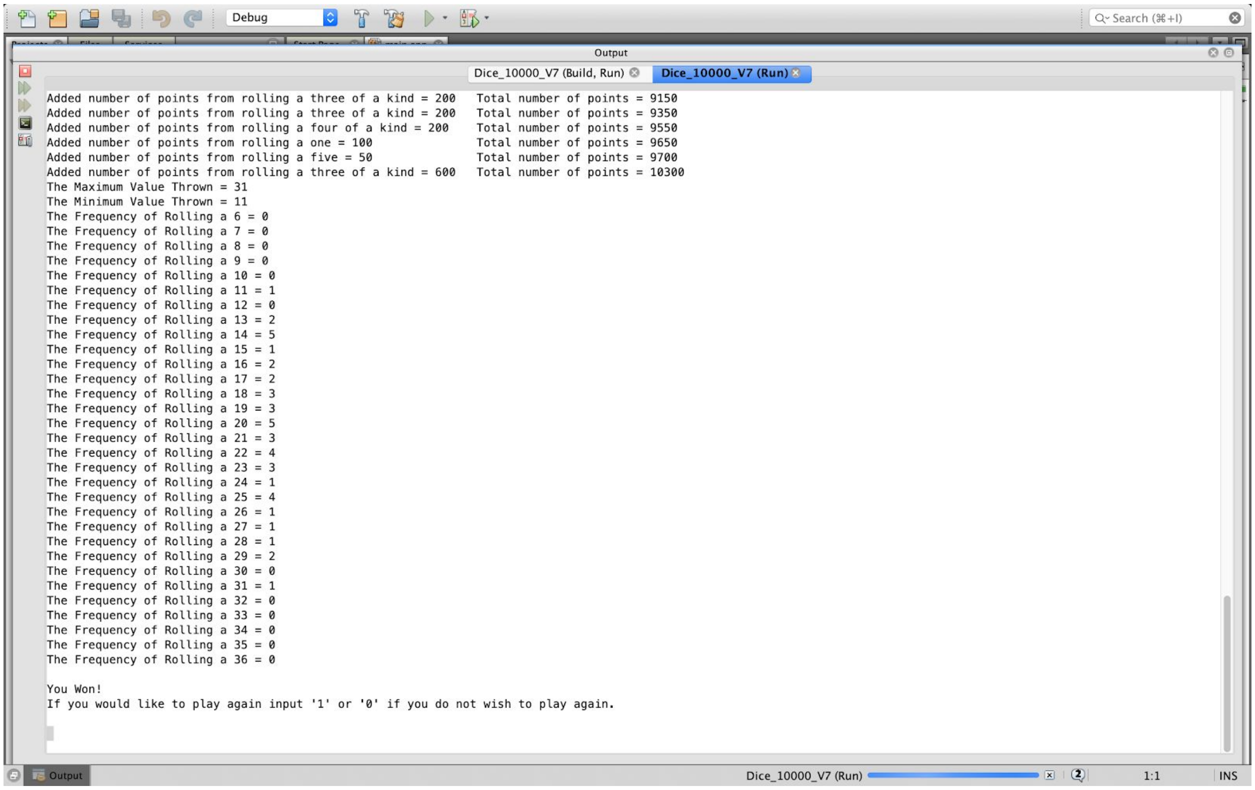
```
Added number of points from rolling a one = 100             Total number of points = 9800
Added number of points from rolling a one = 100             Total number of points = 9900
Added number of points from rolling three ones = 1000       Total number of points = 10900
The Maximum Value Thrown = 29
The Minimum Value Thrown = 11
The Frequency of Rolling a 6 = 0
The Frequency of Rolling a 7 = 0
The Frequency of Rolling a 8 = 0
The Frequency of Rolling a 9 = 0
The Frequency of Rolling a 10 = 0
The Frequency of Rolling a 11 = 1
The Frequency of Rolling a 12 = 0
The Frequency of Rolling a 13 = 0
The Frequency of Rolling a 14 = 1
The Frequency of Rolling a 15 = 2
The Frequency of Rolling a 16 = 1
The Frequency of Rolling a 17 = 0
The Frequency of Rolling a 18 = 5
The Frequency of Rolling a 19 = 2
The Frequency of Rolling a 20 = 5
The Frequency of Rolling a 21 = 3
The Frequency of Rolling a 22 = 1
The Frequency of Rolling a 23 = 1
The Frequency of Rolling a 24 = 5
The Frequency of Rolling a 25 = 1
The Frequency of Rolling a 26 = 0
The Frequency of Rolling a 27 = 3
The Frequency of Rolling a 28 = 1
The Frequency of Rolling a 29 = 1
The Frequency of Rolling a 30 = 0
The Frequency of Rolling a 31 = 0
The Frequency of Rolling a 32 = 0
The Frequency of Rolling a 33 = 0
The Frequency of Rolling a 34 = 0
The Frequency of Rolling a 35 = 0
The Frequency of Rolling a 36 = 0

You Won!
If you would like to play again input '1' or '0' if you do not wish to play again.

0

RUN FINISHED; exit value 0; real time: 57s; user: 0ms; system: 0ms
```

In this example, you can see that the user decided to play the game again. When he decided to play again, the points reset to 0 and a new batch of dice was rolled.



```
The Frequency of Rolling a 16 = 0
The Frequency of Rolling a 17 = 1
The Frequency of Rolling a 18 = 3
The Frequency of Rolling a 19 = 2
The Frequency of Rolling a 20 = 2
The Frequency of Rolling a 21 = 1
The Frequency of Rolling a 22 = 3
The Frequency of Rolling a 23 = 1
The Frequency of Rolling a 24 = 1
The Frequency of Rolling a 25 = 2
The Frequency of Rolling a 26 = 0
The Frequency of Rolling a 27 = 0
The Frequency of Rolling a 28 = 1
The Frequency of Rolling a 29 = 0
The Frequency of Rolling a 30 = 0
The Frequency of Rolling a 31 = 1
The Frequency of Rolling a 32 = 0
The Frequency of Rolling a 33 = 0
The Frequency of Rolling a 34 = 0
The Frequency of Rolling a 35 = 0
The Frequency of Rolling a 36 = 0

You Won!
If you would like to play again input '1' or '0' if you do not wish to play again.

1
Added number of points from rolling a one = 100               Total number of points = 100
Added number of points from rolling a one = 100               Total number of points = 200
Added number of points from rolling a three of a kind = 400   Total number of points = 600
Added number of points from rolling a one = 100               Total number of points = 700
Added number of points from rolling a five = 50               Total number of points = 750
Added number of points from rolling a three of a kind = 600   Total number of points = 1350
Added number of points from rolling a one = 100               Total number of points = 1450
Added number of points from rolling a five = 50               Total number of points = 1500
Added number of points from rolling a one = 100               Total number of points = 1600
Added number of points from rolling a five = 50               Total number of points = 1650
Added number of points from rolling a one = 100               Total number of points = 1750
Added number of points from rolling a one = 100               Total number of points = 1850
Added number of points from rolling a one = 100               Total number of points = 1950
Added number of points from rolling a one = 100               Total number of points = 2050
Added number of points from rolling a one = 100               Total number of points = 2150
Added number of points from rolling a one = 100               Total number of points = 2250
Added number of points from rolling a five = 50               Total number of points = 2300
Added number of points from rolling three ones = 1000         Total number of points = 3300
```

This example shows the end of the second game of 10000. Once again, the game ended when the points added up to 10000. It is clear that this program can be run as many times as the user would like to play, until the user ends the program by inputting "0."



```
Added number of points from rolling a three of a kind = 200   Total number of points = 9150
Added number of points from rolling a three of a kind = 200   Total number of points = 9350
Added number of points from rolling a four of a kind = 200    Total number of points = 9550
Added number of points from rolling a one = 100               Total number of points = 9650
Added number of points from rolling a five = 50               Total number of points = 9700
Added number of points from rolling a three of a kind = 600   Total number of points = 10300
The Maximum Value Thrown = 31
The Minimum Value Thrown = 11
The Frequency of Rolling a 6 = 0
The Frequency of Rolling a 7 = 0
The Frequency of Rolling a 8 = 0
The Frequency of Rolling a 9 = 0
The Frequency of Rolling a 10 = 0
The Frequency of Rolling a 11 = 1
The Frequency of Rolling a 12 = 0
The Frequency of Rolling a 13 = 2
The Frequency of Rolling a 14 = 5
The Frequency of Rolling a 15 = 1
The Frequency of Rolling a 16 = 2
The Frequency of Rolling a 17 = 2
The Frequency of Rolling a 18 = 3
The Frequency of Rolling a 19 = 3
The Frequency of Rolling a 20 = 5
The Frequency of Rolling a 21 = 3
The Frequency of Rolling a 22 = 4
The Frequency of Rolling a 23 = 3
The Frequency of Rolling a 24 = 1
The Frequency of Rolling a 25 = 4
The Frequency of Rolling a 26 = 1
The Frequency of Rolling a 27 = 1
The Frequency of Rolling a 28 = 1
The Frequency of Rolling a 29 = 2
The Frequency of Rolling a 30 = 0
The Frequency of Rolling a 31 = 1
The Frequency of Rolling a 32 = 0
The Frequency of Rolling a 33 = 0
The Frequency of Rolling a 34 = 0
The Frequency of Rolling a 35 = 0
The Frequency of Rolling a 36 = 0

You Won!
If you would like to play again input '1' or '0' if you do not wish to play again.
```